

# NOTES: by Padam Mantry

## HTML: Complete

## CSS: Intermediate

### Introduction to HTML:

HTML = HyperText Markup Language

Title of the html code is displayed on the chrome tab.

Heading will show on the web-page.

Plain text has all the matter but no structure

HTML gives a file its basic structure.

To make some text bigger on a webpage use the various available headline tags available.

Headline tags need a closing tag once opened.

`<h1> <insert your text> </h1>`

To add an enter line use `<br>` tag. This tag is called Line Break tag and does not require an end tag.

**MDN** is a great resource for headings, another great resource is **w3 schools**.

Usage notes can be really relevant while learning.

Another great resource to use is called **devdocs.io**

`<br>` acts as enter on a webpage.

To add big horizontal dash use `<hr>`

To know more about individual elements check out the documentation page from time to time.

devdocs.io

```
<hr size="3" noshade>
```

P.S. Here **noshade** is imp otherwise the changes shall not be reflected.

To centre align your webpage use

`<centre>` tag at the starting and ending of the segment.

Syntax of a comment in HTML is:

```
<!-- <text goes here> -->
```

Conventionally developers name the homepage "index".

## BOILERPLATE:

Boilerplate is used quite frequently throughout programming. It usually means some kind of code template that we can re-use.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

Lets talk about the various terms in the above code.

<!DOCTYPE> declares the document type and tells the browser the version of html this file is using.

<!DOCTYPE html> is standardised for HTML 5 document.

For shortcuts (similar to the boilerplate)

Refer : [docs.emmet.io/cheat-sheet](https://docs.emmet.io/cheat-sheet)

<html> </html> tells the browser that anything written in between those opening and closing tags is the html code.

<title> </title> is used to update the title on the webpage that shall be viewed in the tab lineup.

<meta charset="UTF-8">

The <meta> elements in general give the extra metadata associated with the website. This line particularly however tells the browser that all the data inside is encoded using the charset "utf-8".

Why this particular encoder?

Because it contains almost all international language symbols and every single symbol included in the Unicode character set. <maximum compatibility>

Use <p> </p> to group text into a paragraph

Use <i> </i> to italicize text.

Use <em> </em> or emphasize to italicize text.

Use <em> as a greater priority than <i>.

Reason:

Hume text k baki ke text se different karna hai aur uspe emphasis dena hai, not just style it. (think it like HTML)

Use <strong> </strong> to make text bold. Similar use and explanation with <b>.

```

<> Padam1.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <meta charset="UTF-8">
5  |   <title>Padam Mantry</title>
6  </head>
7  <body>
8  |   <h1>Padam Mantry</h1>
9  |   <p><i>First Year Student at <strong>IIIT Surat</strong>.</i></p>
10 |   <br>
11 |   <p>Extrovert. <br> Minivlogs, Geopolitics and History give me kicks!</p>
12 |   <br>
13 |   <hr size="2" noshade>
14 |   <h3>Padhai Likhai:</h3>
15 |   <p>Welland Gouldsmith School (2008-2019)
16 |   |   <br>Hariyana Vidya Mandir (2019-2021)
17 |   |   <br>IIIT Surat (First Year)
18 |   </p>
19 </body>
20 </html>
21

```

## Bullet Points:

some examples of nested bullet points::

- |               |               |
|---------------|---------------|
| ◦ Milk        | • Milk        |
| ◦ Cheese      | • Cheese      |
| ▪ Blue cheese | ◦ Blue cheese |
| ▪ Feta        | ◦ Feta        |
| ▪ Some        | ▪ Some        |
| ▪             | ▪             |

Syntax:

```

<ul>
  <li>first item</li>
  <li>second item</li>
  <li>third item</li>
</ul>

```

Note ul is for unordered list while u is for underline.

Full form of ul is unordered list and is justified by the name itself.  
ol stands for ordered list. While li for both stands for list item.

The <ul> element is for grouping a collection of items that do not have a numerical ordering, and their order in the list is meaningless. Typically, unordered-list items are displayed with a bullet, which can be of several forms, like a dot, a circle, or a square.

Various variations using list you can do:

Attributes

This element also accepts the global attributes.

reversed

This Boolean attribute specifies that the list's items are in reverse order. Items will be numbered from high to low.

start

An integer to start counting from for the list items. Always an Arabic numeral (1, 2, 3, etc.), even when the numbering type is letters or Roman numerals. For example, to start numbering elements from the letter "d" or the Roman numeral "iv," use start="4".

type

Sets the numbering type:

a for lowercase letters

A for uppercase letters

i for lowercase Roman numerals

I for uppercase Roman numerals

1 for numbers (default)

The specified type is used for the entire list unless a different type attribute is used on an enclosed <li> element.

Using Roman Numeral type

```
<ol type="i">
  <li>Introduction</li>
  <li>List of Grievances</li>
  <li>Conclusion</li>
</ol>
```

Using the start attribute

```
<ol start="4">
  <li>Speedwalk Stu</li>
  <li>Saunterin' Sam</li>
  <li>Slowpoke Rodriguez</li>
</ol>
```

Reversed List examples:

```
<ol reversed>
  <li>ListName</li>
</ol>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Rank in Descending order</h2>
```

```
<ol reversed>
```

```
<li>Kapil</li>
<li>sachin</li>
<li>Will</li>
<li>nikhil</li>
<li>Aakash</li>
<li>Steve</li>
<li>Rahul</li>
<li>Kane</li>
<li>Rohan</li>
<li>John</li>
</ol>
</body>
</html>
```

## Images in HTML:

- Self closing tag

Syntax:

```
 ">
```

The alt portion is important from an SEO point of view.

To use just the name of an image without the full path, make sure that it is located in the same folder as the webpage file.

## Links in HTML: Anchor Tag

Syntax:

Not a self closing tag.

```
<a href= " <link here> " > <Text which contains link> </a>
```

To make links open up in a new tab: (use target="\_blank")

```
<a href= " <link here> " target="_blank" > <Text which contains link> </a>
```

Note: If you have opened some link, it shall appear purple and not blue.

Code up until now:

```
<> Padam1.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <title>Padam Mantry</title>
6  </head>
7  <body>
8    
10   <h1>Padam Mantry</h1>
11   <p><i>First Year Student at <strong><a href="http://www.iiitsurat.ac.in/">IIIT Surat</a></strong>.</i></p>
12
13
14   <hr size="2" noshade>
15   <h3>Padhai Likhai:</h3>
16   <p><ul>
17     <li>Welland Gouldsmith School (2008-2019)</li>
18     <li>Hariyana Vidya Mandir (2019-2021)</li>
19     <li>IIIT Surat (First Year)</li>
20   </ul>
21   <hr size="3" noshade>
22   <h3>Hobbies:</h3>
23   <ol type="i">
24     <li>Dance: <a href="https://www.youtube.com/shorts/1UIgcZ4mCRs" target="_blank">Here's</a>
25       a link to one of the choreography I performed.</li>
26     <li>Listening to music:<a href="https://open.spotify.com/playlist/37i9dQZF1Epr3w5p11N0kh?si=d9415dbb5c74401c" target="_blank">Here's
27       a link to my playlist of on-repeat songs.</li>
28     <li>Vlogging: <a href="https://www.instagram.com/reel/CbJv-6jJcDv2kRQn6MfEOVQL30pg3pFvFVS-Lo0/" target="_blank">Here's</a>
29       a link to one of my MiniVlogs!</li>
30   </ol>
31   </p>
32 </body>
33 </html>
34
```

## Tables in HTML:

Syntax:

```
<table>
  <thead>
    <tr>
      <th colspan="2">The table header</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>The table body</td>
      <td>with two columns</td>
    </tr>
  </tbody>
</table>
```

thead means Table Head.

td means table data

Everything between <td> and </td> are the content of the table cell.

tr stands for table row.

Each table row starts with a <tr> and end with a </tr> tag.

Sometimes you want your cells to be headers, in those cases use the <th> tag instead of the <td> tag:

By default, the text in <th> elements are bold and centered, but you can change that with CSS.

```
<table>
  <tr>
    <th>Name</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Jill Smith</td>
    <td>50</td>
  </tr>
</table>
```

Note:

Usually the header cell contents are written within the <thead> tag.  
But they can also be written within the <tr> tag, with their <th> tag instead of <td>

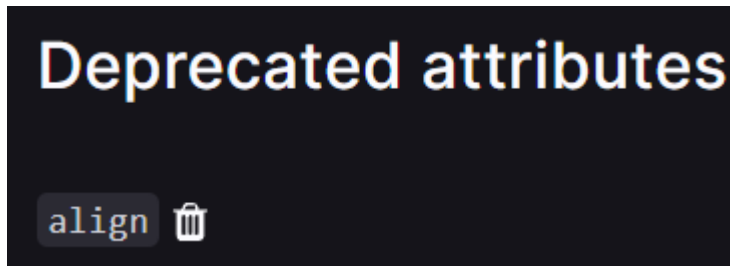
## HTML Table Tags:

Tag	Description
<table>	Defines a table
<th>	Defines a header cell in a table
<tr>	Defines a row in a table
<td>	Defines a cell in a table
<caption>	Defines a table caption
<colgroup>	Specifies a group of one or more columns in a table for formatting
<col>	Specifies column properties for each column within a <colgroup> element
<thead>	Groups the header content in a table
<tbody>	Groups the body content in a table
<tfoot>	Groups the footer content in a table

```
<table>
  <tr>
    <th>Company</th>
    <th>Contact</th>
    <th>Country</th>
  </tr>
  <tr>
    <td>Alfreds Futterkiste</td>
    <td>Maria Anders</td>
    <td>Germany</td>
  </tr>
  <tr>
    <td>Centro comercial Moctezuma</td>
    <td>Francisco Chang</td>
    <td>Mexico</td>
  </tr>
</table>
```

Note: Don't confuse table data contents with lists.

Within the <tr> tag, even though contents are written from top to down, they shall be filled in from left to right.



On the mdn developer docs, you might find some attributes that have a garbage sign or are called deprecated attributes.

What does that mean?

It means that it has been downgraded or is no longer recommended. Remember that HTML is for giving the structure to the WebPage and not styling.

Although it's not recommended you may go on and use it.

Note: HTML Tables by default do not have columns. The arrangement of text gives it a look that columns are there.

HTML Tables can also be used to structurize your webpage overall not just for presenting tabular data.

Note: To give just an outline border in html table without bordering individual cells, use this syntax:

```
<table frame="box" cellspacing="10">
```

To shorten the Horizontal Rule and align it use the below syntax and modify accordingly.

```
<hr size="3" width="400" noshade align="left">
```

## Forms in HTML:

Syntax:

```
<form>
.
.
form elements
.
.
</form>
```



When autocomplete is on, the browser automatically complete values based on values that the user has entered before. Tip: It is possible to have autocomplete "on" for the form, and "off" for specific input fields, or vice versa.

<https://www.w3schools.com> › tags › att\_form\_autocomplete

## HTML form autocomplete Attribute - W3Schools

type=" "	Description
text	Defines a one-line text input field
password	Defines a one-line password input field
submit	Defines a submit button to submit the form to server
reset	Defines a reset button to reset all values in the form.
radio	Defines a radio button which allows select one option.
checkbox	Defines checkboxes which allow select multiple options form.
button	Defines a simple push button, which can be programmed to perform a task on an event.
file	Defines to select the file from device storage.
image	Defines a graphical submit button.

The name attribute is used to reference elements in a JavaScript, or to reference form data after a form is submitted. Note: Only form elements with a name attribute will have their values passed when submitting a form.

<https://www.w3schools.com> › tags › att\_input\_name

## HTML input name Attribute - W3Schools

Lets dive into some syntax:

```
<input type="text" name="firstname">
```

Here type="" needs to be specified as the input type while the name= "" with the description of the data we are passing.

I don't understand the name much right now so pls adjust.

Radio allows only one option to be checked rather than checkbox more than one can be selected.

Refer this for more help on HTML form input:

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/input>

```

<form action="Padam1.html" method="post" enctype="">
  <label>Your Name:</label>
  <input type="text" name="name" value=""><br>
  <label>Your email:</label>
  <input type="email" name="email" value=""><br>
  <label>Your message:</label><br>
  <textarea name="" id="" cols="30" rows="10"></textarea><br>
  <label>Send me all updates via email:</label>
  <input type="checkbox" name="response" id="">
  <input type="submit" name="">
</form>

```

Output:

Your Name:

Your email:

Your message:

Send me all updates via email: ☐

Check out my initial website design files here:

<https://padammantry04.github.io/CV/>


## CSS:

### Cascading Style Sheets

Cascading Style Sheets (CSS) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

Basically styles up our html page.

To change the background image of your webpage:

```
<body style="background-color:  yellow;">
```

Now to get a hold of all the various keywords for colors refer here:

[https://developer.mozilla.org/en-US/docs/Web/CSS/color\\_value](https://developer.mozilla.org/en-US/docs/Web/CSS/color_value)

Refer here for colour palette curated by graphic designers:

<https://colorhunt.co/>

How to use these colors?


Just replace yellow in the above code syntax with the code of the color you want (make sure u use a hashtag)

Now CSS code:

If we want to want to modify an element that occurs multiple times through out the web page, we do not want to go down the cumbersome lane of changing the style attributes of every single element multiple times.

So we use CSS.

Note this has to be used in the head section

```
<style>
  body {
    background-color:  #5EE6EB;
  }
</style>
```

Declare what you need to change then open it like a function declaration:

Within it write the attribute you want to modify and after ":" write how you would like the attribute to be.

To make websites mobile friendly too, use percentages in height and width type elements instead of hard-coded pixel values.

The border-style property may be specified using one, two, three, or four values.

- When one value is specified, it applies the same style to all four sides.
- When two values are specified, the first style applies to the top and bottom, the second to the left and right.
- When three values are specified, the first style applies to the top, the second to the left and right, the third to the bottom.
- When four values are specified, the styles apply to the top, right, bottom, and left in that order (clockwise).

CSS Code up until now:

```

css files > # styles.css > ...
1  body {
2      background-color: #DFF6FF;
3  }
4  hr{
5      background-color: transparent;
6      border: none;
7      height: 0px;
8      border-style:dotted none none;
9      width: 10%;
10     border-color: lightgrey;
11     border-width: 8px;
12     align-items: center;
13 }
14 h1{
15     color: #66BFBF;
16 }
17 h3{
18     color: #66BFBF;
19 }
20

```

Refer this link to find all resources centralised and divided according to chapters:  
<https://www.appbrewery.co/p/web-development-course-resources>

Priority if CSS attributes are referred to at various locations:

### Inline CSS> Internal CSS> External CSS

This gives us an idea that we can apply global CSS rules while still be able to tailor make specific changes to individual pages.

## CSS Syntax:

```

selector{
    property:value;
}

```

We talked about the most simple type of CSS tag selectors. Those which target the html attributes. We can also modify styling by targeting using class and id selectors.

Main purpose:

- They allow more specific changes than general changes like img or h1.

Class and id attributes:

The .class selector selects elements with a specific class attribute.

```
.<class name> {  
  css declarations;  
}
```

HTML elements can also refer to more than one class.

```
<p class="center large">This paragraph refers to two classes.</p>
```

Class vs id selectors:

Syntax of id selectors:

- Declared within the HTML attribute.
- #<id name>{<code>}

Difference between class and id selectors and their various usage:

The same class can be used for various html attributes, but an id cannot be used for various html attributes. Id is for more unique things which need individual styling and cannot be grouped with any while class shall be used when individual styling is required for some html and not all of that type.

Pseudo class:

Hover selector makes the element behave a bit differently than it would when the cursor or mouse is hovering over it.

Syntax:

```
<html attribute>:hover {  
  css declarations;  
}
```

Favicon:

Usually started as favourite icon, these are images or icons that show right beside your title of the webpage.

How can you make one for your website?

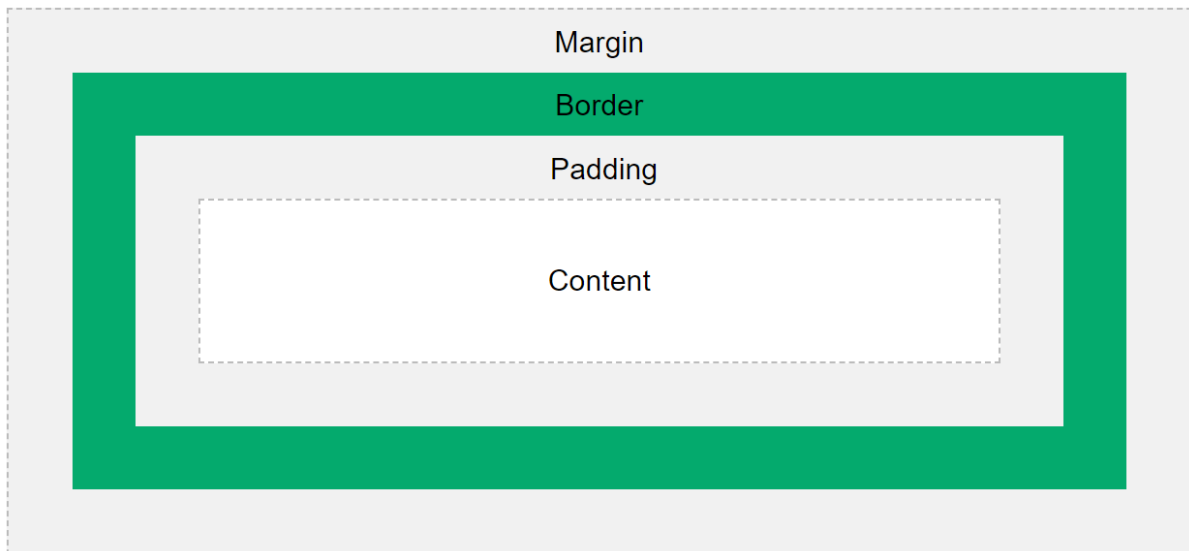
Go to [favicon.cc](https://favicon.cc) and create a sample favicon, then head over to code and add a link in the head of your html page and instead of putting relation as stylesheet, put rel="icon".

Div:

Content division element, allows us to divide or split our content into various containers.

## The Box Model of styling that CSS follows:

The CSS box model is essentially a box that wraps around every HTML element. It consists of: margins, borders, padding, and the actual content. The image below illustrates the box model:



**Content** - The content of the box, where text and images appear.

**Padding** - Clears an area around the content. The padding is transparent.

**Border** - A border that goes around the padding and content.

**Margin** - Clears an area outside the border. The margin is transparent.

Other way to specify:

```
{padding:<top><right><bottom><left>}
```

## CSS Display Property:

The CSS Display property has four different values

### Block

In this type of display the element takes up the whole width of the webpage, essentially blocking the whole width from any other element sitting next on the left or on the right.

Ex: paragraphs, lists, headers, divs, forms

The `<span>` tag is an inline container used to mark up a part of a text, or a part of a document.

The `<span>` tag is easily styled by CSS or manipulated with JavaScript using the class or id attribute.

The `<span>` tag is much like the `<div>` element, but `<div>` is a block-level element and `<span>` is an inline element.

### Inline

Unlike block elements, inline elements only take up as much space is sufficient for the particular element.

Ex: span, images, anchor-tags

Inline elements

Pro:

They are much more flexible and allow horizontal placing.

Con:

The width cannot be changed.

You can modify display settings in the CSS stylesheet as  
display:<display style>

Inline Block

These have the properties of both inline and block.

We consider images as inline block elements

None

It hides a particular element and treats it as if it never existed.

## How to make elements invisible?

visibility:hidden

hides the element, but it still takes up space in the layout.

display:none

removes the element from the document. It does not take up any space.

## FONTS:

There are two main families.

Serif and San-Serif

Serif

This comes as more serious, authoritative and also a bit older.

Applications:

Letterheads/Magazines for architectural companies, etc

Difference between Traditional and Modern Serif:

Take a look at the difference between the thickest and the thinnest part of a letter.  
As the font becomes more and more modern, the difference amplifies.

San-Serif:

Simple, Friendly, Sensible, Straight forward.

Readability and Legibility becomes very important.

Just because you can use many fonts doesn't mean it makes your webpage more stylish, It makes it more chaotic. Try sticking to two or max three fonts while designing a webpage/poster.

Choosing the two fonts:

Mood

Find fonts that have similar moods.

Time

Find fonts that have similar time era

Contrast

Contrast between the serifness of the fonts to generate more interest.

Weights:

Change the weights between the heading and the body to generate more interest.

## CSS Positioning

Some basic rules:

#1: Content is everything.

The content is the first thing that determines how large things get displayed and what the height and width will be.

#2: Order of the element comes from Code:

The elements are placed on the web-page in order of them in the code.

#3: Children sit on top of their parents:

If I place something within the domain of other element as a div or a body tag then the element gets formatted onto the previous parent element.

Position: